GLOBAL OPTIMIZATION OF *n*-MANEUVER, HIGH-THRUST TRAJECTORIES USING DIRECT MULTIPLE SHOOTING

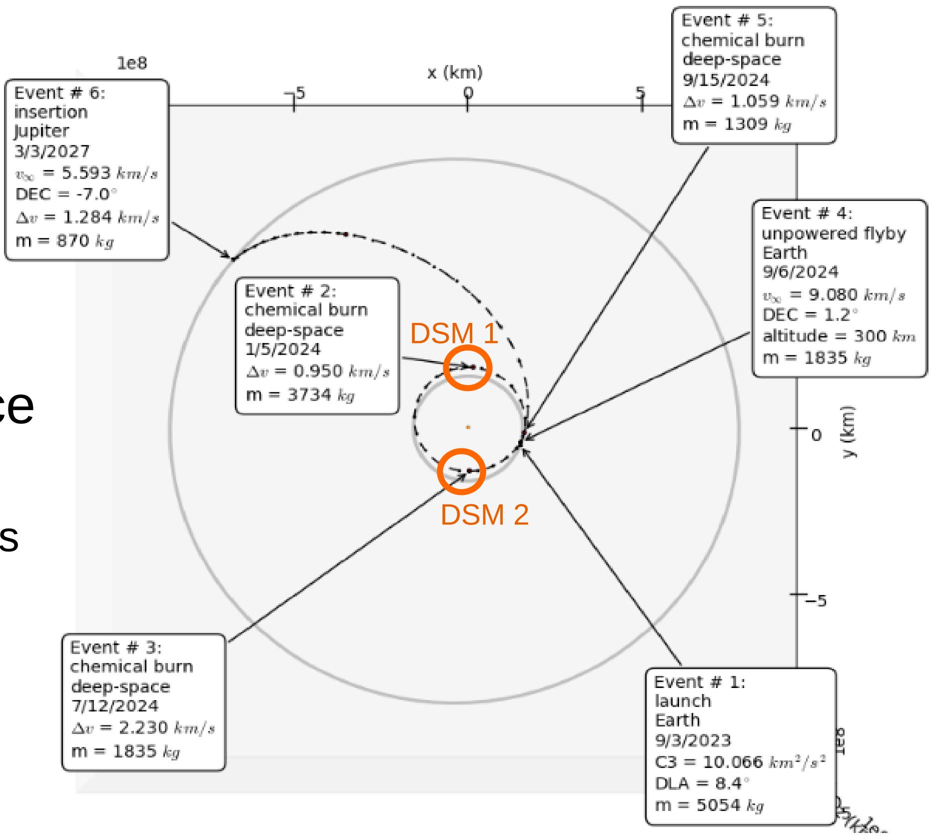
Matthew A. Vavrina, a.i. solutions
Jacob A. Englander, NASA GSFC
Donald H. Ellison, University of Illinois

February 15, 2015

n-Impulse, Gravity-Assist Trajectory Optimization

- Optimal number of maneuvers not known a priori
  - Deep-space maneuvers (DSMs) frequently improve performance
  - Two (or more) DSMs can be optimal for tightly/uniquely constrained trajectory legs
- Missions interested in best performance possible, i.e., the global optimum
  - Can enable cost- or time-constrained missions
  - Often interested in more than 1 objective, e.g., max. delivered mass & min. TOF

Optimal EEJ trajectory w/ 2 DSMs on 1 leg



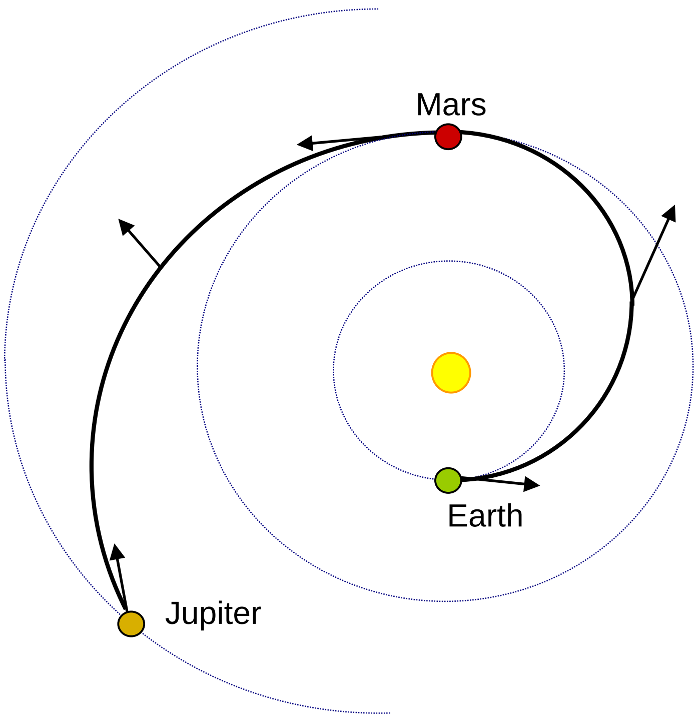
- Problem challenges:
  - Grid searches can be intractable for any number & variety of DSMs
  - Optimization requires initial guess
  - Maneuvers and gravity assists can create highly sensitive optimization problems

Objective

Globally optimize chemical propulsion based trajectories with an arbitrary number of maneuvers & gravity assists

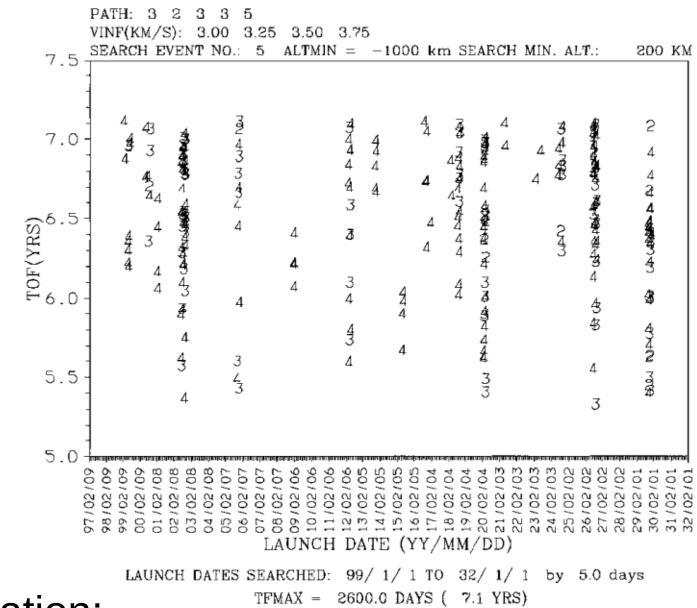
Method should be:

- Automated
- No requirement of a user-defined initial guess
- Able to search broad design space
- Efficient (medium-fidelity is appropriate)
- Capable of handling multiple objectives



Prior Approaches to Global Search

Example STOUR plot (TOF vs. LD)
Petropoulos et al., "Trajectories to Jupiter…" JSR, 2000



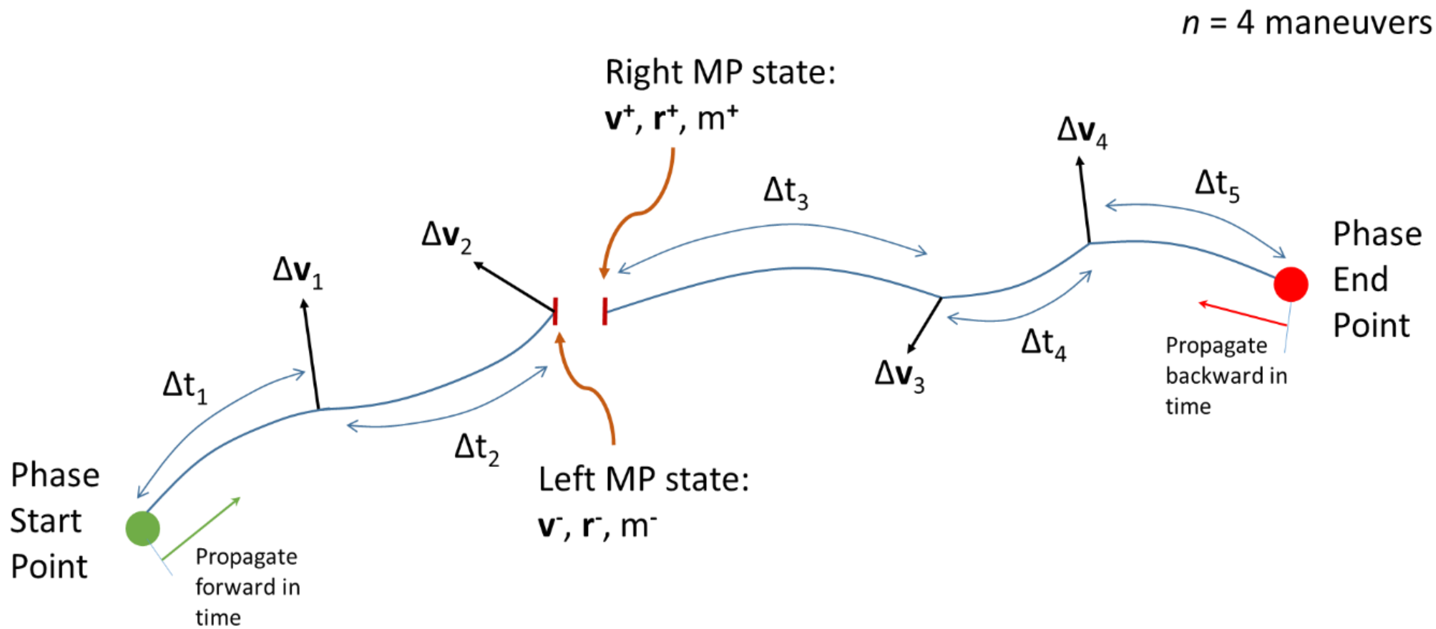
- Grid search
  - Lambert scans over range of event dates & flyby bodies
  - Strategies developed to include one maneuver per leg
    - Patel and Longuski → STOUR (Purdue, JPL)
    - Lantukh and Russell (UT-Austin)
    - Maneuvers limited to specific type v-infinity leveraging or broken plane maneuver
- Stochastic search
  - Strategically sample design space
  - Typically use a direct, Lambert-based trajectory formulation:
    - Vary departure date, time to DSM, & DSM components
    - Propagate forward to DSM point
    - Solve Lambert problem to subsequent body, repeat for all legs
    - Guaranteed feasibility for unconstrained problems
  - Not limited in maneuver type, but problems are frequently very sensitive to initial guess
  - Most approaches not capable of more than 1 DSM
  - Global-local hybrid scheme can improve efficacy

MGA*n*DSMs Trajectory Transcription

Multiple gravity assist with *n* deep space maneuvers using shooting scheme *s*

→ MGA*n*DSM*s*

- Aim for robustness & efficiency with an arbitrary number of DSMs with a direct formulation
- Nonlinear programming problem
- Employs forward/backward shooting similar to Byrnes & Bright (CATO)
- Nominally Kepler propagation between maneuvers
- Analytic match point (MP) constraints
- User specifies *n*, optimizer reduces 1 or more DSM magnitude to zero if <*n* optimal



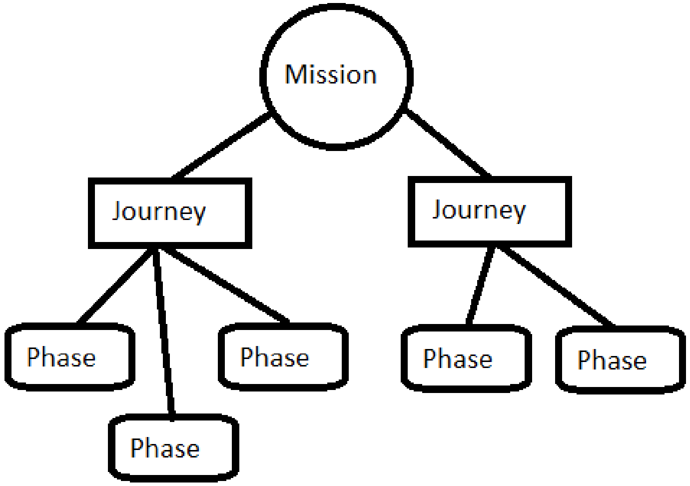
Core Optimization Variables
Δt_{phase}
Δt_1
$\Delta t_2, \ldots \Delta t_n$
Δt_{n+1}
$\Delta\mathbf{v}_1, \Delta\mathbf{v}_2, \ldots, \Delta\mathbf{v}_n$
$\mathbf{v}_{initial}$
$\mathbf{v}_{final}$

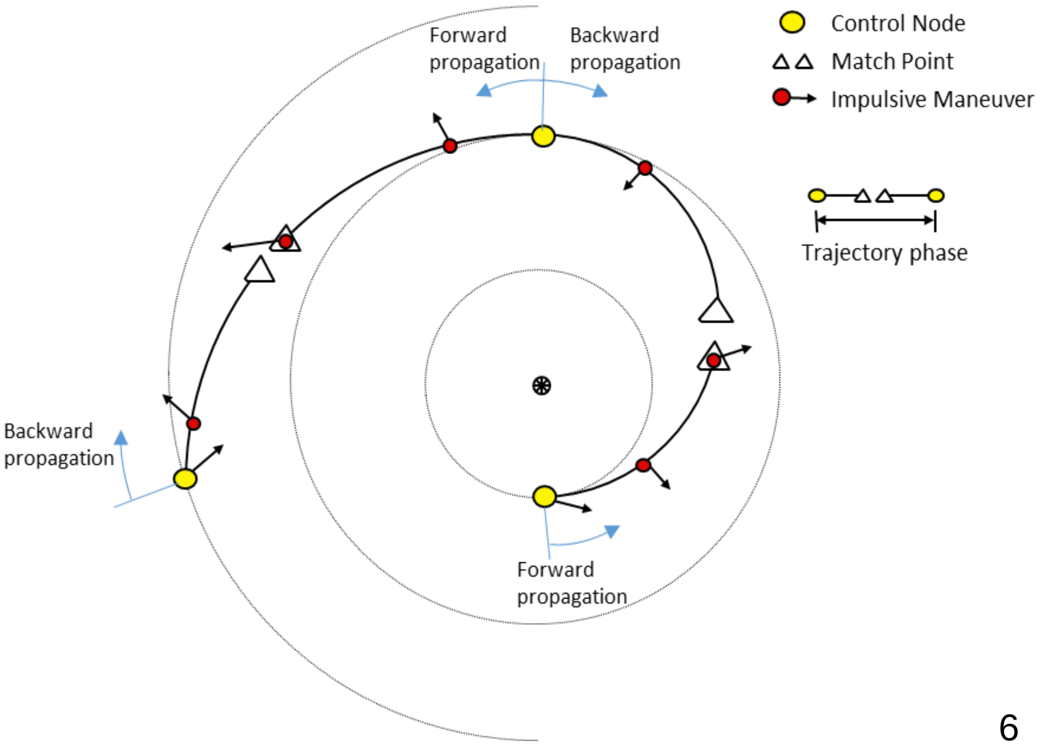
Match point continuity constraint:

$$\mathbf{c}_{mp} = \left[r_x^+ - r_x^-,\, r_y^+ - r_y^-,\, r_z^+ - r_z^-,\, v_x^+ - v_x^-,\, v_y^+ - v_y^-,\, v_z^+ - v_z^-,\, m^+ - m^-\right]^T = \boldsymbol{\varepsilon}^T$$

MGA*n*DSMs Mission Formulation

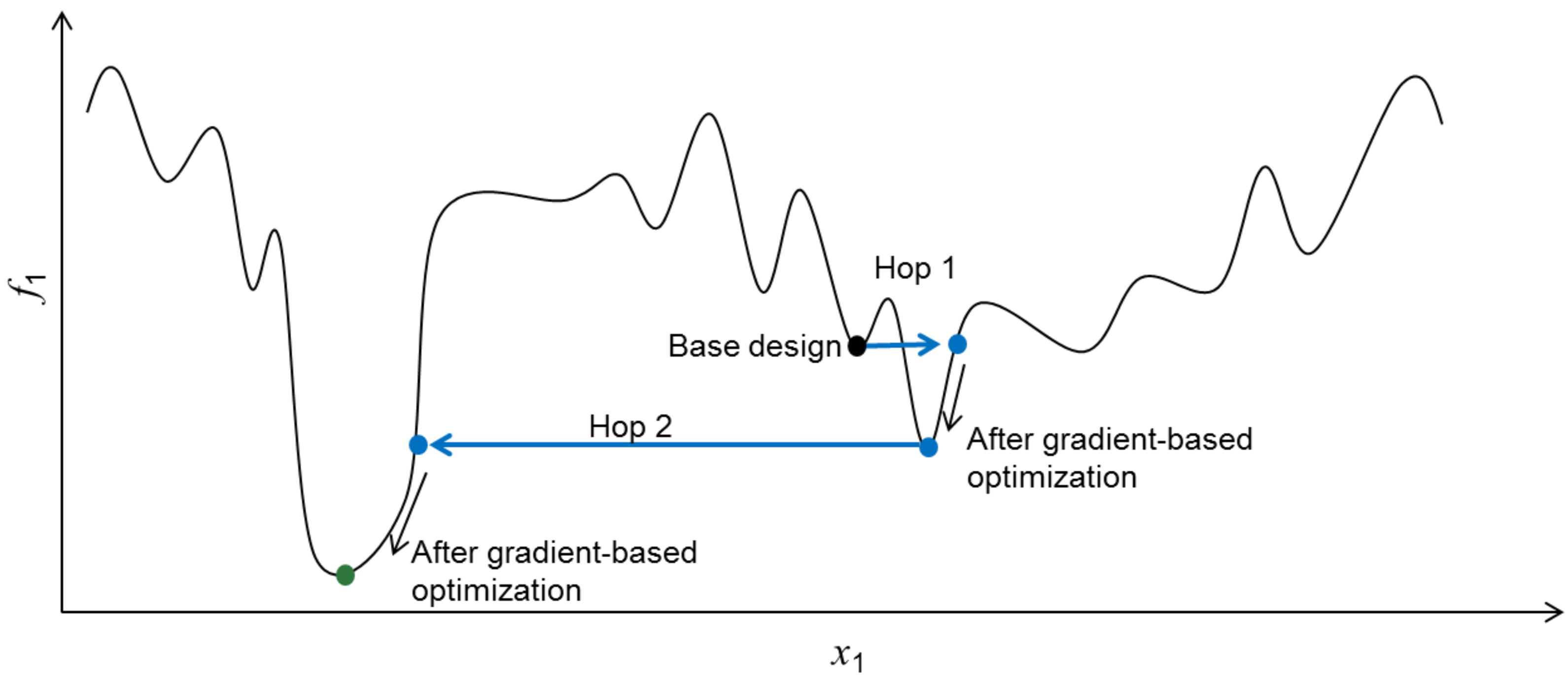
- Multiple gravity assists accommodated in mission structure
- Allows for variation of flyby bodies at control nodes of transcription
  - Journeys start & end a user-required bodies/states
  - Journeys can be composed of multiple phases with variable flyby bodies to improve performance (“null gene” approach)
- Zero sphere of influence patched conics
  - Flyby constraints ensure physically realizable gravity assist
- Flexible to numerous mission constraints





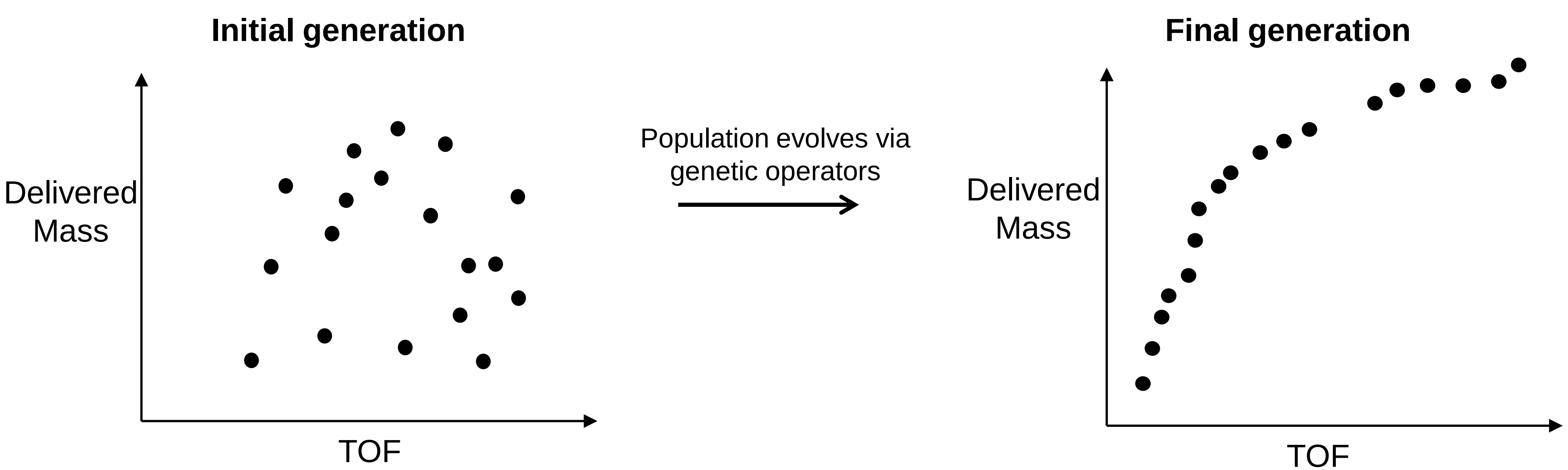
MGA*n*DSMs Global Optimization

- Combine monotonic basin hopping (MBH) & sequential quadratic programming (SQP) → MBH+SQP
- Stochastic, global search scheme
- No initial guess required
- Adept at multi-modal problems w/ clustered local minima
- Stochastic "hops" evaluated from base solution



Multi-objective Optimization

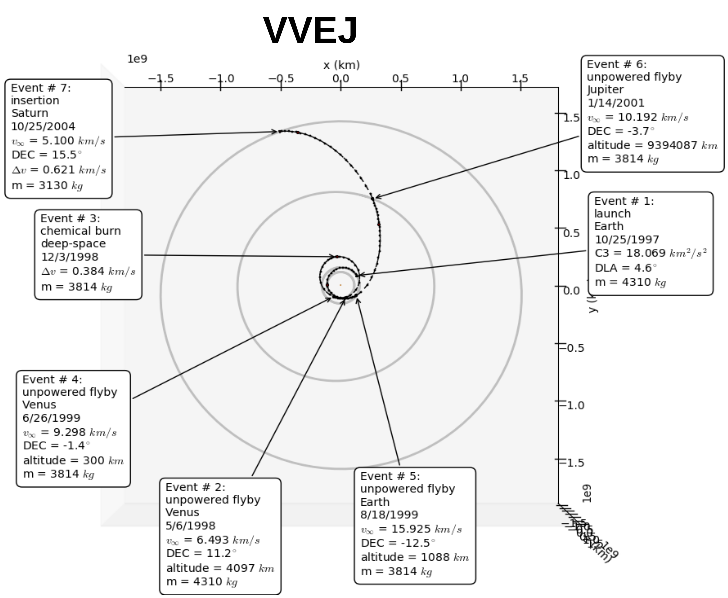
- MGA*n*DSM*s* structured within multi-objective hybrid optimal control algorithm (discrete & continuous variables)
- Multi-objective genetic algorithm (GA) serves as outer loop systems optimizer around direct-method inner loop trajectory optimizer
  - Outer loop: non-dominated Sorting Genetic Algorithm II (NSGA-II) searches over discrete mission parameters, defining trajectory problem for inner loop
    - Variables include: Flyby body, target body, launch vehicle, launch C3, launch epoch
  - Inner loop: MBH+SQP solves trajectory problem & establishes obj. func. values
  - Generates representation of Pareto front (optimal tradeoff between objectives)



Cassini Test Case Comparison

- Established global optimum serves a test case for comparison of MGA*n*DSMs & a Lambert-based transcription (MGADSM*k*); one DSM allowed
- MGA*n*DSMs identified global optimum in 10 out of 10 cases
- MGADSk identified solution w/in 50 m/s of global optimum, but not the global optimum (1.004 km/s) after an 8 hour run time
- Order of magnitude improvement in time to best solution
  - Median time is 25x faster

Run Number	MGA*n* DSMs Time to Identify Global Optimum (minutes)	MGADSM*k* Time to Identify Best Solution (minutes)	MGADSM*k* Time Identify Solution within 50 m/s of Global Optimum (minutes)	MGADSM*k* Best ΔV (km/s)
1	5.3	223.8	220.3	1.025
2	7.9	140.9	32.9	1.043
3	8.8	260.8	149.3	1.026
4	12.8	74.3	74.3	1.014
5	63.1	178.1	122.0	1.020
6	4.3	295.4	112.0	1.034
7	5.0	471.3	465.7	1.039
8	6.3	382.9	328.5	1.049
9	9.4	80.6	58.4	1.033
10	38.4	63.6	63.6	1.038
Mean	**16.1**	**217.2**	**162.7**	**1.032**
Median	**8.4**	**200.9**	**117.0**	**1.033**



Example Problem: Multi-objective Optimization to Jupiter

- Evaluated MGA*n*DSMs multi-objective capability on a multiple gravity assist to Jupiter with up to 2 DSMs per phase
- Flyby bodies are varied with any combination up to 5 bodies
- Three objectives: maximize $\log_{10}$(final mass), minimize TOF, minimize Jupiter arrival C3
  - Pareto front: 3D surface of equally optimal solutions
- Population size of 256 and the MBH+SQP inner loop is allowed to run for 40 minutes

Outer-loop Design Variables

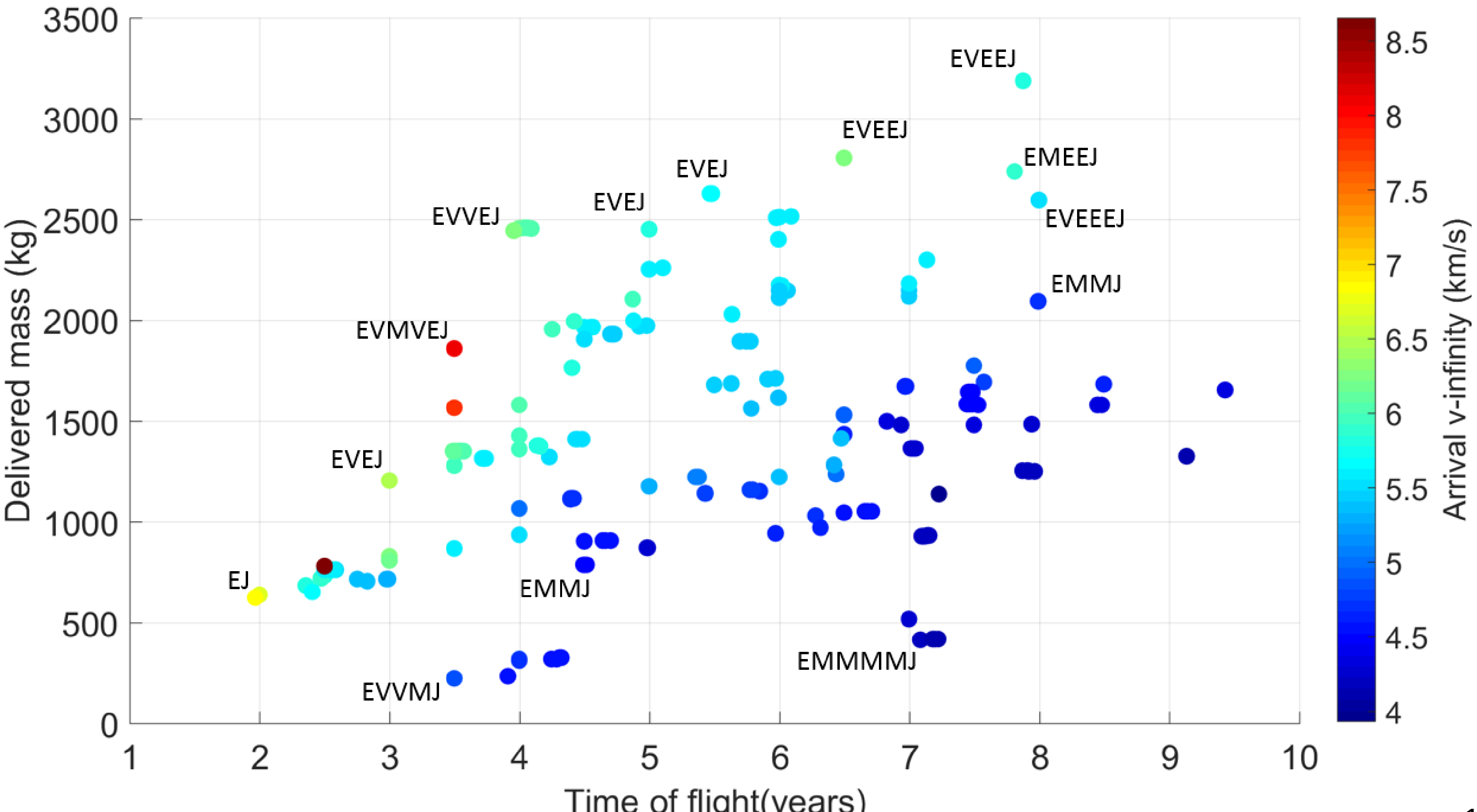
Design Variable	Value	Resolution
Launch window open epoch	{1/1/2021, 1/1/2022, 1/1/2023, 1/1/2024}	1 year
Flyby body	{Venus, Earth, Mars, null, null, null}	n/a
Flight time	[730, 3467.5] days	182.5 days

Common Mission Parameters

Description	Value
Launch window	365.24 days
Launch declination	[-28.5, 28.5] deg
Launch vehicle curve	Atlas V, 551
Chemical I_{sp}	320 s
Jupiter arrival date	Determined by optimizer
Jupiter insertion orbit semi-major axis	10,054,900 km (140.6 R_J)
Jupiter insertion orbit eccentricity	0.911
Maximum number of DSMs	2
Inner-loop objective function	Max: $\log_{10}$(final mass)
Inner-loop run time	40 minutes

Best Non-Dominated Front

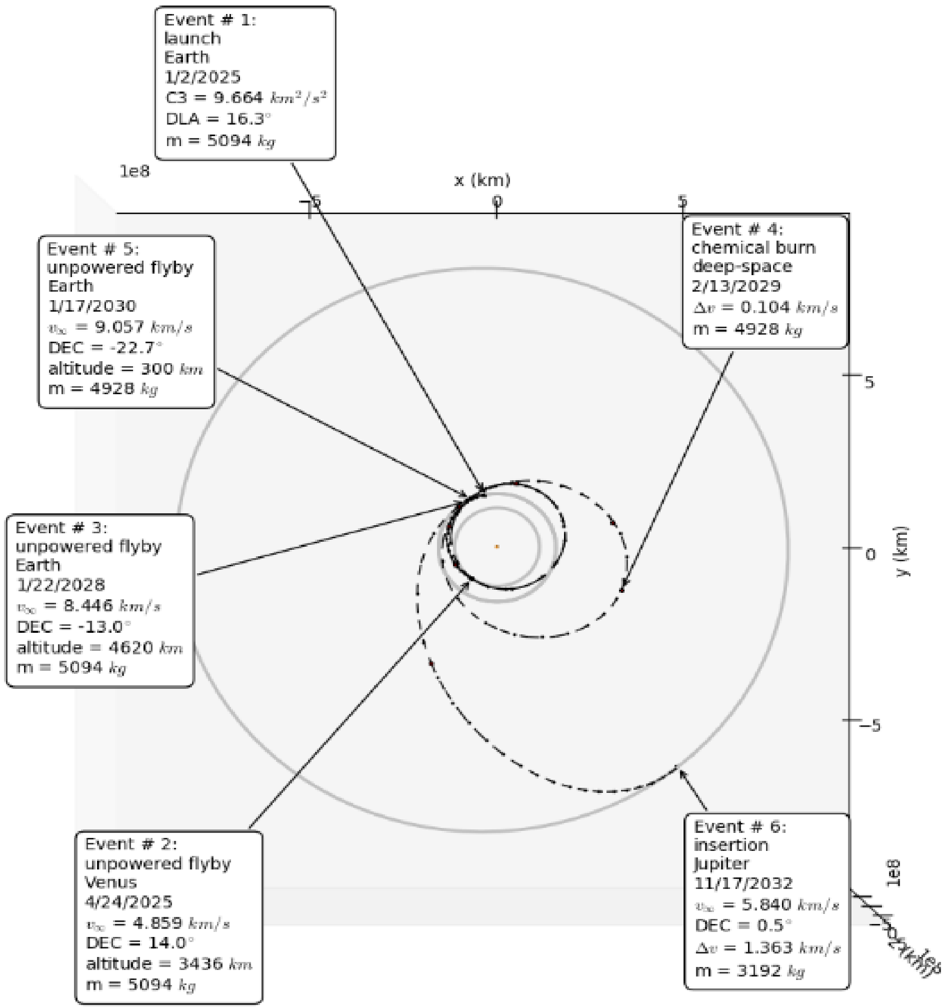
- Representation of 3D Pareto front generated after 100 generations
- 3.5 days of run time on 64-core processor



Example Optimal Jupiter Trajectories

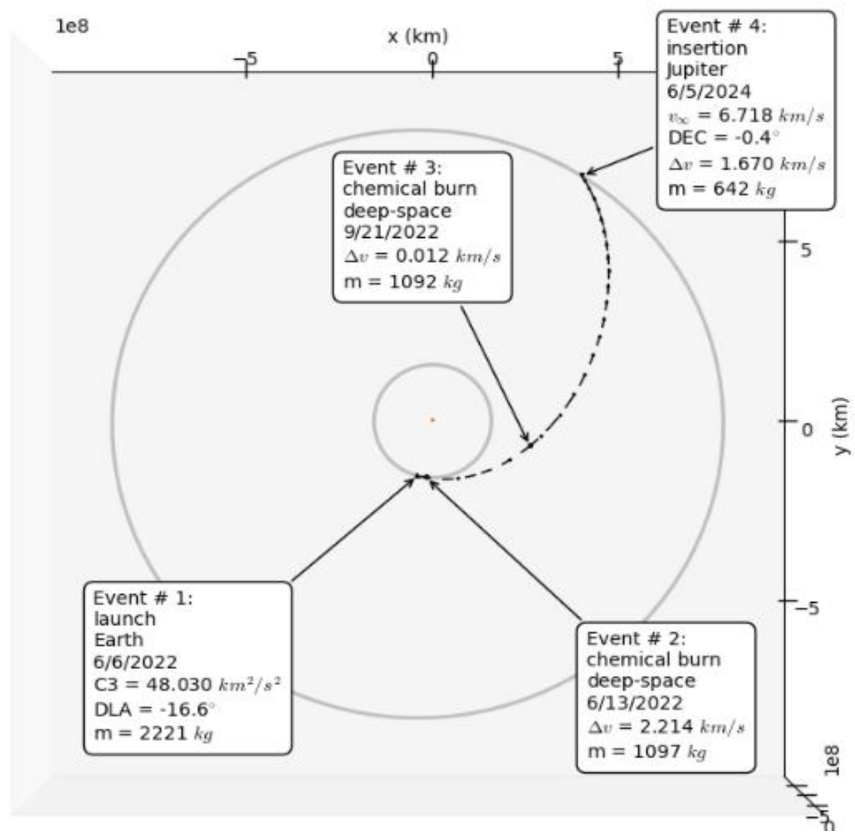
Highest Delivered Mass Trajectory

- EVEEJ sequence
- Delivered mass: 3192 kg
- TOF: 7.9 years
- Arrival C3: 34.1 km/s



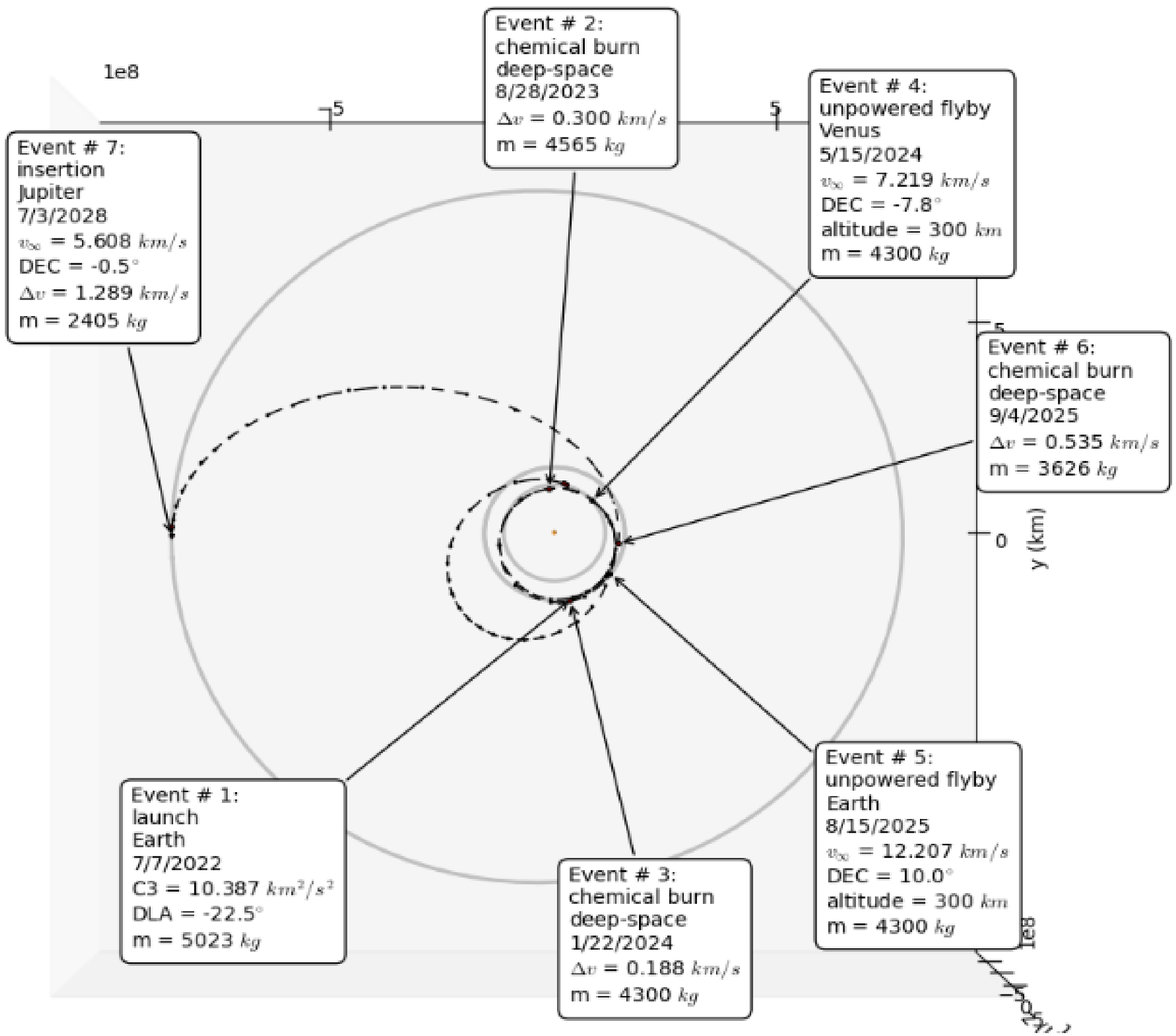
Shortest TOF Trajectory

- Direct EJ (2 DSMs optimal)
- Delivered mass: 642 kg
- TOF: 2.0 years
- Arrival C3: 45.1 km/s



Optimal Jupiter Trajectory with 2 DSMs in 1 Phase

EVEJ sequence



Conclusions

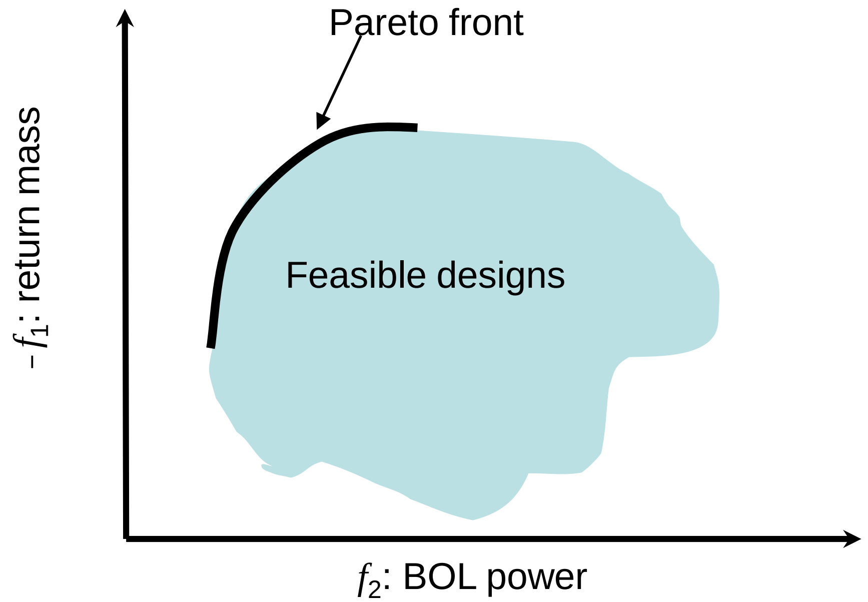
- Developed high-thrust trajectory optimization transcription for any number of maneuvers between flybys
- Multiple shooting framework & analytic derivatives provide robustness and enable outer-loop efficiency
- Formulation allows for an automated, global search without a user-supplied initial guess
- Capability to generate Pareto-optimal solutions using multi-objective hybrid optimal control algorithm
- General applicability to almost any variety of interplanetary mission
  - Flexible to unique trajectory/mission constraints
- Large problems can become computationally tractable

Backup

Multi-objective Optimization

- Want to optimize any number of mission design metrics
  - e.g., payload mass, TOF, arrival C3
  - Often coupled & competing
  - Fully map mission trade-offs between optimal solutions

- Optimize multiple objectives simultaneously
  - Entire set of optimal solutions
  - Goal: generate representation of Pareto front
  - Traditionally use repetitions of single objective technique



NSGA-II

- Develops globally-optimal Pareto solutions using non-dominated sorting
  - Conducts stochastic, global search with population of designs

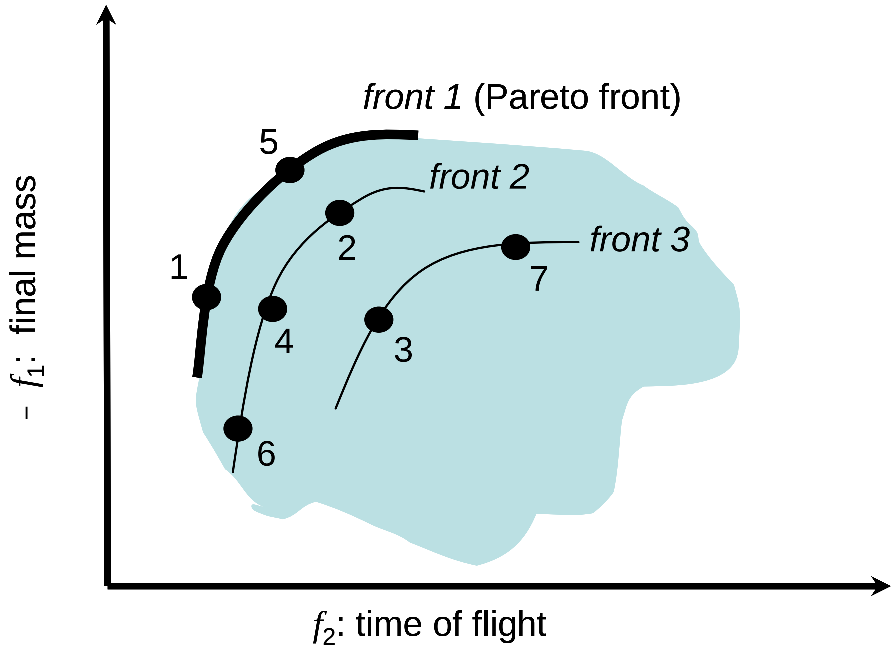
- Fitness assignment based on "nearness" to Pareto front
  - $\mathbf{x}_1$ dominates $\mathbf{x}_2$ if:

$$\forall p : f_p(\mathbf{x}_1) \leq f_p(\mathbf{x}_2) \qquad p = 1,2,\ldots,n_{obj}$$

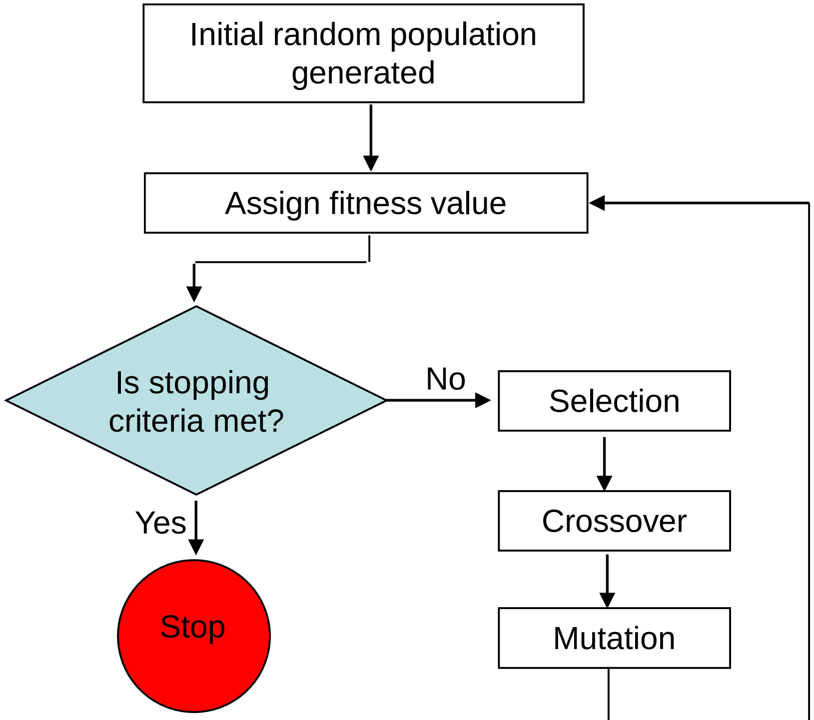
and

$$\exists p : f_p(\mathbf{x}_1) < f_p(\mathbf{x}_2) \qquad p = 1,2,\ldots,n_{obj}$$

- If neither design dominates other, they are non-dominant
- Non-dominated sorting:
  - Assign fitness based on design's non-dominated front
  - Designs closer to Pareto front get more mating opportunities



Genetic Algorithm



- Models Darwinian evolution
  - Mimic natural selection & reproduction
- Searches with population of designs
- Globally search design space
- No initial guess required